



TEKNILLINEN TIEDEKUNTA

TEKOÄLYN SOVELLUKSIA KONETEKNIIKASSA

Kiia Kokko

KONETEKNIIKAN TUTKINTO-OHJELMA

Kandidaatintyö

Joulukuu 2020

TIIVISTELMÄ

Tekoälyn sovelluksia konetekniikassa

Kiia Kokko

Oulun yliopisto, Konetekniikan tutkinto-ohjelma

Kandidaatintyö 2020, 34 s.

Työn ohjaaja yliopistolla: Yrjö Louhisalmi

Työn tavoitteena on esitellä tekoälyn eri sovelluksia konetekniikassa. Sovelluksia edeltää tekoälyyn liittyvän teorian läpikäynti, jotta sovelluksissa käytettäviä tekoälyn menetelmiä pystytään ymmärtämään. Teoriaosassa käydään läpi tekoälyn eri osa-alueet ja keskeisimmät käsitteet. Työssä tarkastellaan myös, mitä uutta tekoäly tuo konetekniikkaan ja millä keinoin. Työssä on käytetty apuna aiempien aiheeseen liittyvien tutkimusten tuloksia, tieteellisiä artikkeleita sekä kirjallisuutta. Tuloksina löydettiin useita erilaisia tekoälyn sovelluksia konetekniikassa ja koneoppimista käytettiin lähes jokaisessa tapauksessa. Tekoäly mahdollistaa tulevaisuudessa entistä tiiviimmän yhteistyön ihmisten ja koneiden välillä.

Asiasanat: Tekoäly, koneoppiminen, konetekniikka

ABSTRACT

Applications of artificial intelligence in mechanical engineering

Kiia Kokko

University of Oulu, Degree Programme of Mechanical Engineering

Bachelor's thesis 2020, 34 p.

Supervisor at the university: Yrjö Louhisalmi

The aim of this bachelor's thesis is to present different applications of artificial intelligence in mechanical engineering. Applications are preceded by a review of the theory of artificial intelligence in order to understand the methods of artificial intelligence used in applications. The theoretical part reviews the different aspects of artificial intelligence and the most important concepts. The thesis also examines what new artificial intelligence brings to mechanical engineering and by what means. The results of previous researches, scientific articles and literature have been used in this work. Several different applications of artificial intelligence in mechanical engineering were found and machine learning was used in almost every case. Artificial intelligence will enable even closer cooperation between human and machine in the future.

Keywords: Artificial intelligence, machine learning, mechanical engineering

ALKUSANAT

Työn tarkoituksena on perehtyä tekoälyyn ja sen sovelluksiin ja menetelmiin konetekniikassa. Työ on toteutettu ajanjaksolla 8.9.-17.11.2020. Haluaisin kiittää Oulun yliopiston kandintyötäni ohjannutta Yrjö Louhisalmea opastuksesta ja neuvoista koko kandintyön tekemisen aikana ja antamalla mahdollisuuden tähän mielenkiintoiseen aiheeseen.

Oulu, 17.11.2020

Kiia Kekko

Työn tekijä

SISÄLLYSLUETTELO

TIIVISTELMÄ

ABSTRACT

ALKUSANAT

SISÄLLYSLUETTELO

MERKINNÄT JA LYHENTEET

| | |
|--|----|
| 1 JOHDANTO | 7 |
| 1.1 Työn tavoite | 7 |
| 1.2 Työn rajausta | 7 |
| 2 MITÄ TEKOÄLY ON? | 8 |
| 2.1 Yleistä | 8 |
| 2.2 Heikko ja vahva tekoäly | 9 |
| 2.3 Symbolinen ja konnektionistinen tekoäly | 10 |
| 3 KONEOPPIMINEN | 11 |
| 3.1 Yleistä | 11 |
| 3.2 Ohjattu oppiminen | 12 |
| 3.3 Ohjaamaton oppiminen | 14 |
| 3.4 Vahvistusoppiminen | 15 |
| 3.5 Neuroverkot ja syväoppiminen | 16 |
| 3.6 Konenäkö | 18 |
| 4 TEKOÄLYN SOVELLUKSET KONETEKNIKKASSA | 21 |
| 4.1 Betonin puristuslujuuden ennustaminen koneoppimisella | 21 |
| 4.2 Konenäkö laaduntarkastuksessa ja kunnossapidossa - yleistä | 23 |
| 4.3 Konenäkö tarkastusta varten - tapaustutkimus | 23 |
| 4.4 Materiaalien löytö ja suunnittelu koneoppimisella | 25 |
| 4.5 Pyörivien koneiden vianmääritys tekoälyn avulla | 27 |
| 4.6 Mitä uutta tekoäly tuo konetekniikkaan | 28 |
| 5 YHTEENVETO | 30 |
| LÄHDELUETTELO | |

MERKINNÄT JA LYHENTEET

| | |
|------|-------------------------------|
| AI | Artificial Intelligence |
| CAD | Computer-Aided Design |
| DT | Decision Tree |
| k-NN | k-Nearest Neighbors Algorithm |
| ML | Machine Learning |
| NN | Neural Network |
| RF | Random Forest |
| SVM | Support Vector Machine |

1 JOHDANTO

1.1 Työn tavoite

Työn aiheena on tekoäly ja sen sovellukset konetekniikassa. Tavoitteena on perehtyä tekoälyyn ja sen osa-alueisiin. Työssä käydään läpi erilaisia konetekniikan käytännön sovelluksia, jossa käytetään tekoälyä hyväksi eri tavoilla. Tekoäly on taas vahvasti käytössä teknologian aloilla ja etenkin tuotteiden tuotannossa sen käytöllä voidaan optimoida toimitusketjua ja nopeuttaa tutkimusta ja kehitystä. Tekoälyä pystytään soveltamaan monissa eri tuotannon prosesseissa ja konetekniikan osa-alueissa ja niiden kehitys on jatkuvaa. Työssä tutkitaan tapauksia, jossa tekoälyä käytetään konetekniikan alalla. Koneoppiminen on tekoälyn osa-alue, joka mahdollistaa järjestelmien oppimisen ja parantaa sovellettavien kohteiden toimintoja automaattisesti opitun tiedon pohjalta. Koneoppimisella pystytään toteuttamaan monenlaiset rutiininomaiset tehtävät tuotannossa ilman ihmisen ohjausta. Koneoppimisella pystytään myös kehittämään uutta ja parempaa tekniikkaa, kun kerätään tarpeeksi dataa nykyisestä tilanteesta.

1.2 Työn rajaus

Työssä tarkastellaan tekoälyä, sovelluksia ja siihen liittyviä käsitteitä. Työssä esitellään myös tekoälyn haaroja yleisesti, jotta lukijalla olisi hyvät lähtökohdat saada ymmärrystä mahdollisista sovelluksista. Tapaustutkimukset esitetään kattavasti toteutuksesta lopputuloksiin saakka. Huomiotta jätetään eri koneoppimisen mallien matemaattiset esitykset ja perustelut. Lopputuloksia ei perustella numeerisin keinoin vaan keskitytään tarkastelemaan käytännönläheisemmin eri tapauksia.

2 MITÄ TEKOÄLY ON?

2.1 Yleistä

Tekoäly eli Artificial Intelligence (AI), on tietokoneohjelma tai tietokone, joka kykenee suorittamaan älykkäitä toimintoja ja ihmismäistä ajattelua. Tekoälylle ei ole täysin yksiselitteistä termiä, jolla sitä voitaisiin kuvata mahdollisimman kattavasti ja samalla tuoda sen ominaisuudet esille. Antti Merilehto kuvailee teoksessaan tekoälyn olevan koneen suorittamaa toimintaa, joka ihmisen tekemänä olisi älykästä. Hän luettelee tekoälylle ominaisia toimintoja, joita ovat muun muassa päättely, oppiminen, ennakointi, päätöksenteko, näkö ja kuulo (Merilehto, 2018, s. 18). Toisen näkökulman mukaan ohjelmistoa voidaan pitää tekoälynä, jos ohjelmisto on autonominen ja adaptiivinen. Tällä tarkoitetaan sitä, että ohjelmisto pystyy suorittamaan tehtäviä ilman jatkuvaa ohjausta sekä sen suorituskyky paranee oppimisen perusteella (FabricAI, 2020). Kumpikin määritelmä tuovat esille tekoälyn ominaisuudeksi päättelyn sekä oppimisen. Nämä ovatkin edellytys sille, että jotakin toimintaa voidaan kutsua tekoälyksi, sillä pelkkä data ei vielä ole älykkyyttä. Tätä dataa on pystyttävä käsittelemään ja opettelemaan, jonka jälkeen voidaan tehdä päättelyitä ja näin ollen päätöksiä sen pohjalta. Jos kone kykenee edelliseen toimintaan, voidaan puhua, että siinä on tekoälyä.

Eräs tapa määritellä tekoäly on jakaa sille esitetyt määritelmät neljään eri luokkaan ryhmiteltynä, taulukon 1. mukaisesti. Stuart Russellin ja Peter Norvigin (2010) luoman taulukon yläosa liittyy ihmisen ajatteluun ja päättelyyn ja alemmat käyttäytymiseen. Järjestelmän voidaan sanoa olevan rationaalinen, jos se tekee oikeita asioita sille annetun tiedon perusteella. (Russell & Norvig 2010, Pietikäinen & Silvén, 2019, s.17 mukaan)

Taulukko 1. Tekoälyn ryhmittely (mukaillen Pietikäinen & Silvén 2019)

| | |
|--|--|
| Thinking Humanly eli ajatella ihmisen lailla | Thinking Rationally eli ajatella rationaalisesti |
| Acting Humanly eli toimia ihmisen lailla | Acting Rationally eli toimia rationaalisesti |

Alan Turing esitteli seuraavan määritelmän, jolla koneen älykkyyttä voitaisiin käytännössä testata. Oletuksena on, että tila on jaettu seinällä, jonka toisella puolella on ihmiskyselijä ja toisella puolella joko ihmisvastaaja tai tekoälyvastaaja eli tietokone. Testi toteutetaan niin, että kommunikointi tapahtuu esimerkiksi tietokoneiden avulla. Mikäli testaaja päätyy siihen lopputulokseen, että hän on keskustellut ihmisen kanssa, vaikka todellisuudessa onkin käynyt keskustelua tekoälyvastaajan kanssa, tällöin Turingin määritelmän mukaan koneella on älykkyyttä. Testille on asetettu vaatimukset sen läpäisemiseksi ja tällöin koneen on pystyttävä seuraaviin asioihin: luonnollisen kielen käsittelyyn kommunikointia varten, kykyyn tallentaa mitä kone tietää ja oppii tietämään, automaattiseen päättelyyn tallennetun tiedon avulla ja uusien johtopäätösten tekoon sekä koneen on käytettävä koneoppimista pystyäkseen ilmaisemaan ja yleistämään malleja toimintaympäristöstään. (Russell & Norvig 2010, Pietikäinen & Silven, 2019, s.17-18 mukaan)

2.2 Heikko ja vahva tekoäly

Tekoäly voidaan jakaa kahteen eri kategoriaan, heikkoon ja vahvaan tekoälyyn. Heikko eli sovellettu tekoäly on tehokkaasti toimiva järjestelmä, joka jäljittelee ihmisen älykkyyttä mutta kykenee suorittamaan ainoastaan yhtä ennalta määrättyä tehtävää kerrallaan. Yhtäläisyyksien löytäminen isosta joukosta dataa tai jonkin ennalta määrätyn asian tunnistaminen kuvasta ovat esimerkkejä heikon tekoälyn tehtävistä. Kaikki nykyiset tekoälyn sovellukset perustuvat heikkoon tekoälyyn. (CGI, 2020) Heikon tekoälyn avulla pystytään myös tunnistamaan tekstiä, puhetta ja eri kieliä. Lisäksi shakin ja muiden pelien pelaaminen on mahdollista. (Pietikäinen ja Silvén, 2019, s. 23) Vahva eli yleinen tekoäly on koneen kykyä jäljitellä inhimillistä älykkyyttä kokonaisvaltaisesti. Tällöin kone kykenee käyttämään erilaisia taustatietoja päätöksiä suunnitellessaan ja tehdessään. Esimerkiksi luonnolliseen ihmisen ja koneen väliseen keskusteluun tarvittaisiin vahvaa tekoälyä, kun halutaan keskustelun muistuttavan ihmisten välistä keskustelua. Tällöin koneen on ymmärrettävä keskustelun sisältö kokonaan, mihin heikko tekoäly ei vielä pysty. Nykyiset tekoälyratkaisut ovat vielä kaukana vahvasta tekoälystä.

Superäly voidaan saavuttaa siinä vaiheessa, jos ihminen kehittää itseään älykkäämmän koneen, joka taas itsenäisesti kehittää yhä vahvempia ratkaisuja. Tällöin voidaan puhua teknologisesti singulariteetista. (Pietikäinen ja Silvén, 2019, s. 23)

2.3 Symbolinen ja konnektionistinen tekoäly

Symbolisessa tiedonkäsittelyssä rakennetaan suuria tietokantoja ja sääntöjä pohjautuen johonkin ongelmaan. Symbolinen tekoäly perustuu päättelyiden ja abstraktioiden käyttöön ongelmanratkaisussa. Tieto pyritään esittämään konkreettisina seikkoina ja sääntöinä, jotka esittävät toimenpiteiden ehtoja ja vaikutuksia, yleensä merkkijonoina. Mahdolliset haasteet tulevat tarvitun tietämyskannan hankkimisesta ja rakentamisesta. Vahvuutena on se, että esitystavat ja päättelyketjut sekä niiden tulokset ovat helposti ihmiselle ymmärrettävässä muodossa. Tällöin virheelliset toiminnot pystytään jäljittämään ja korjaamaan. Tietämyksen esittämisessä symbolisesti vaikuttavia tekijöitä ovat syntaksi, semantiikka ja ontologia. Syntaksi luo säännöt ihmisen ja koneen ymmärrettävissä olevien kuvausten laadintaan. Semantiikka luo syntaktisia sääntöjä noudattaen määritellyt merkitykset. Ontologia kuvaa käsitteet ja niiden väliset semanttiset suhteet. (Pietikäinen ja Silvén, 2019, s. 46-48)

Esimerkiksi kuvantunnistuksessa symbolinen tekoäly ei ole riittävän fiksu. Jos kuvasta halutaan tunnistaa esimerkiksi tietty kissa, tehdään ohjelma, joka perustuu sääntöihin ja vertailee alkuperäistä kuvaa kissasta sille syötettyyn kuvaan ja syötteenä tulee tulos, onko kuvassa sama kissa vai ei. Tämä toimii siihen saakka, kun ohjelmalle syötetään täydellinen kopio alkuperäisestä kuvasta. Kuitenkin vähänkin erilainen kuva kissasta antaa negatiivisen tuloksen. Esimerkiksi, jos ohjelmalle syötetään kuva kissasta eri kuvakulmasta, ohjelma menee rikki eikä tunnista kissaa. (Dickson, 2019)

Dataohjattujen konnektionistinen tiedonkäsittely on keinotekkoisten hermoverkkojen eli neuroverkkojen perusta. Dataohjatulla tiedonkäsittelyllä tarkoitetaan menettelyä, jossa päätökset perustuvat kerättyyn ja analysoituun tietoon. Neuroverkot koostuvat erilaisista laskentamenetelmistä, joita yhdistetään toisiinsa ja joita voidaan opettaa. Erona symboliseen tekoälyyn, dataohjatussa tekoälyssä dataa ei muodosteta merkkijonoihin. Järjestelmä rakentaa oman esitystapansa. Kun sille syöttää kuvan, se voi tunnistaa kuvissa olevia kohteita. Tämä luo mahdollisuuden opettamisen automatisointiin. Haasteena konnektionistisessa tekoälyssä on se, että virheellisille toiminnoille on äärimmäisen vaikea löytää selityksiä, sillä järjestelmän esitystapa on ihmiselle vaikea ymmärtää. Luotettavuutta pyritään vahvistamaan syöttämällä oppimisvaiheessa suuria määriä dataa, jotta kaikki mahdolliset vaihtoehdot tulisivat läpikäytyä. (Pietikäinen ja Silvén, 2019, s. 48)

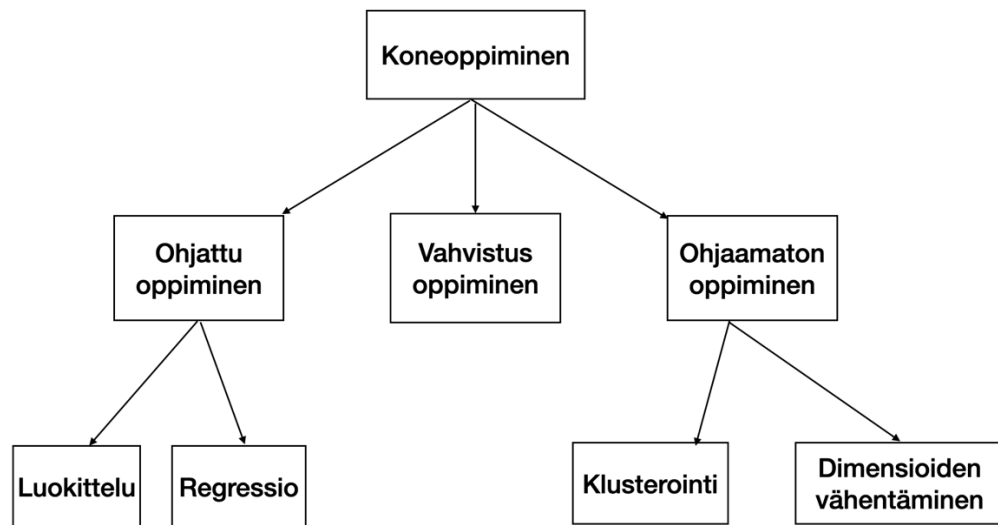
3 KONEOPPIMINEN

3.1 Yleistä

Koneoppiminen eli Machine Learning (ML) on yksi tekoälyn osa-alue. Koneoppimisessa käytetään dataa oppimiseen ja luokitteluun. Koneelle ei ole määritelty toimintaohjetta jokaista erillistä tilannetta varten. Koneoppiminen käyttää algoritmeja, jotka oppivat käytössä olevasta datasta jatkuvasti. Näin on mahdollista kuvailla käytettävissä olevaa dataa paremmin ja sen avulla ennustamaan lopputuloksia. Datan määrä vaikuttaa tuloksen tarkkuuteen. Mitä enemmän dataa on hallussa, sitä tarkemman lopputuloksen kone saa. (Merilehto, 2018, s. 19-27)

Koneoppiminen voidaan jakaa kolmeen osa-alueeseen, riippuen datan ja ratkaistavien ongelmien luonteesta. Ohjattua oppimista on esimerkiksi tapaus, jossa tehtävänä on tunnistaa liikennemerkki ja vastaus voi olla yksinkertaisesti ”kyllä” tai ”ei”. Ohjaamattomassa koneoppimisessa koneen tehtävä on löytää datasta rakenne, esimerkiksi samankaltaisten tapausten ryhmiä, jonka avulla data voidaan esittää tärkeimpien muuttujien avulla. Vahvistusoppimista voidaan käyttää tilanteissa, joissa palaute siitä oliko tietty ratkaisu oikea vai väärä, tulee mahdollisesti viiveellä. Se sopii käytettäväksi peleihin, joissa lopputulos ratkeaa vasta pelin loputtua. (Reaktor & Helsingin yliopisto, 2018)

Kuvassa 1 koneoppiminen voidaan jakaa oppimisen tyylin perusteella ohjattuun-, ohjaamattomaan- sekä vahvistusoppimiseen. Ohjattu oppiminen jaetaan edelleen datan tyyppin mukaan: luokittelu ja regressio. Ohjaamattomalla oppimisella puolestaan data ryhmitellään, klusterointiin ja dimensioiden vähentämiseen, ominaisuuksien perusteella.



Kuva 1. Koneoppimisen menetelmät (mukaillen Tuominen & Neittaanmäki 2019).

3.2 Ohjattu oppiminen

Suurin osa käytännön koneoppimisesta on ohjattua oppimista (eng. Supervised learning). Ohjatussa oppimisessa on syötemuuttujat (X) ja tulostemuuttuja (Y) ja algoritmia käytetään kartoitustoiminnon oppimiseen syötteistä tulosteiksi. Tavoitteena on arvioida kartoitus niin hyvin, että kun syötetään uusi syötetieto, voidaan ennustaa tulostemuuttujat tästä datasta. (Brownlee, 2016) Ohjatun oppimisen ongelmat voidaan edelleen jakaa kahteen ryhmään, regressioon ja luokitteluun. Ero näiden kahden välillä on, että regressiossa tunnistetaan mallia datan rakenteelle ja luokittelussa tunnistetaan rakennetta, johon data mallissa kuuluu. (Pietikäinen ja Silvén, 2019, s. 72)

Luokittelussa tuloste muuttuja on kategoriaa esimerkiksi ”punainen” tai ”sininen” tai ”sairaus” ja ”ei sairautta”. Regressio-ongelmassa on olemassa luokiteltuja näytteitä. Lääketieteelliset diagnoosit ovat esimerkki luokitteluongelmasta, jossa muuttujina ovat potilaan oireet, lääkärin havainnot ja laboratorioanalyysien tulokset. Järjestelmälle opetetaan valmiiksi kategorisoidut näytteet ja tämän jälkeen järjestelmä kykenee tunnistamaan uusien näytteiden luokkia niiden piirteiden perusteella. (Pietikäinen ja Silvén, 2019, s. 73)

Regressio-ongelmassa on olemassa syötteitä vastaavia mittauksia. Esimerkiksi regressiota on, kun tulostemuuttuja on reaalinen arvo kuten ”eurot” tai ”paino”. Pietikäinen ja Silvén esittävät yhdeksi esimerkiksi käytetyn auton hinnan määrittämiseksi, jolloin syötteinä ovat merkki, vuosimalli, kilometrit ja selitettävänä muuttujana on hinta. Kumpaakin menetelmään liittyy kuitenkin ylioppimisen ja alioppimisen riski. Esimerkkinä regressiopuu, joka voi kärsiä ylioppimisesta, jos data-avaruutta viedään liian pitkälle ja jokainen lehtisolmujen regressiomalli ennustaa vain yhden opetusnäytteen pohjalta ja antaa hyviä tuloksia vain opetusnäytteen kanssa identtiselle datalle. Toinen esimerkki löytyy neuroverkoista, jossa opetusdatan liian vähäinen määrä johtaa ylioppimiseen. Alioppimisella tarkoitetaan tilannetta, jossa saatu malli ei riitä jäljittelemään datan rakennetta. Tyypillinen alioppimisesta kärsivä kohde on lineaarinen regressio, jossa selittävien ja selitettävien muuttujien yhteys on epälineaarinen ja tällöin ennusteiden tarkkuus jää heikoksi. (Pietikäinen ja Silvén, 2019, s. 75)

Ohjatun oppimisen luokittelualgoritmeja ovat Naive Bayes, k:n lähimmän naapurin menetelmä eli k-nearest neighbors (k-NN) sekä tukivektorikone eli Support Vector Machine (SVM).

Naive Bayesin algoritmi perustuu Bayesin 1. lauseeseen (1) ja sen avulla käsitellään muuttujien todennäköisyysjakaumia tietojoukossa ja ennustetaan tiettyyn arvoon kuuluvan vastemuuttujan todennäköisyyttä, kun otetaan huomioon uuden esiintymän attribuutit. Sillä voidaan tutkia tapahtuman todennäköisyyttä kaikkien aiempaan tapahtumaan liittyvien tapahtumien ennakkotietojen perusteella. Esimerkiksi todennäköisyyttä, että talon hinta on korkea, voidaan paremmin arvioida, jos tunnetaan sen ympärillä olevat tilat verrattuna arvioon, joka on tehty ilman tietoa talon sijainnista. Tämän luokittelijan etuna on, että se vaatii pienen määrän opetusdataa luokituksen edellyttämien parametrien arvioimiseksi. (Gupta, 2017)

$$P(A|B) = \frac{P(B|A) * P(A)}{P(B)} \quad (1)$$

missä $P(A|B)$ = ehdollinen todennäköisyys, että tapahtuma A tapahtuu, kun otetaan huomioon, että B on tapahtunut
 $P(A), P(B)$ = A:n ja B:n todennäköisyys toisistaan riippumatta
 $P(B|A)$ = ehdollinen todennäköisyys, että tapahtuma B tapahtuu, kun otetaan huomioon, että A on tapahtunut

k-NN-algoritmin oletuksena on, että samanlaiset asiat ovat lähellä toisia. k-NN tarvitsee luokitteluun arvon k, etäisyysmitan ja runsaan opetusaineiston. Jotta tietojen kannalta oikea k:n arvo saadaan valittua, suoritetaan KNN algoritmi useita kertoja erilaisilla k-arvoilla ja valitaan k, joka vähentää kohtaamien virheiden määrää samalla säilyttäen algoritmin kyky tehdä tarkkoja ennusteita, kun sille annetaan dataa, jota se ei ole aiemmin nähnyt. Kun k:n arvoa kasvatetaan, ennustukset vakautuvat keskiarvon laskun ansiosta ja siten todennäköisyys tarkempiin ennusteisiin kasvaa, tiettyyn pisteeseen asti. Liian suuri arvo tuottaa virheitä. K:n arvon laskeminen puolestaan muuttaa ennustukset vähemmän vakaiksi. (Harrison, 2018)

SVM:n eli tukivektorikonealgoritmin tavoitteena on löytää hypertaso N-ulotteisesta avaruudesta (N kuvaa ominaisuuksien lukumäärää), joka luokittelee datapisteet selvästi. Tavoitteena on löytää taso, jolla on suurin marginaali eli suurin etäisyys molempien luokkien datapisteiden välillä. Hypertason kummallekin puolelle putoavat datapisteet voidaan osoittaa eri luokkiin. Hypertason koko riippuu myös ominaisuuksien määrästä. Tukivektorit ovat datapisteitä, jotka ovat lähempänä hypertasoa ja vaikuttavat hypertason sijaintiin ja suuntaukseen. Käyttämällä näitä tukivektoreita maksimoidaan luokittelijan marginaali. (Gandhi, 2018)

3.3 Ohjaamaton oppiminen

Ohjaamattomassa oppimisessä on syötetty data mutta ei vastaavaa tulostemuuttujaa. Tavoitteena ohjaamattomassa oppimisessä on mallintaa datan taustarakenne tai jakauma datan lisäoppimista varten. Verrattuna ohjattuun oppimiseen, ohjaamattomassa ei ole oikeaa ratkaisua ongelmalle eikä ohjelmalla ole opettajaa. Algoritmit tekevät itse

havainnot ja niiden pohjilta esittävät tietyn rakenteen datasta. (Brownlee, 2016) Ohjaamaton oppiminen voidaan jakaa kahteen ratkaisumalliin, dimensioiden vähentämiseen ja klusterointiin, jonka valinta riippuu olemassa olevasta etukäteistiedosta.

Klusteroinnissa halutaan saada selville luontainen ryhmittely datalle, esimerkiksi ryhmittelemällä asiakkaat heidän ostokäyttäytymisensä mukaan. Kun käytetään klusterointia, tarjolla on tietoa kategorioiden lukumäärästä, toisin kuin dimensioiden pudotusongelmissa. Klusteroinnin periaatteena on jakaa näytedata ryhmiin niin, että ryhmän sisällä olevat kriteerit ovat lähempänä toisiaan kuin eri ryhmien väliset. (Duda ym. 2001 Pietikäinen ja Silvén mukaan s. 92)

Dimensionaalisuus eli ulottuvuus on muuttujien, ominaisuuksien tai piirteiden määrä datajoukossa. Esimerkiksi dimensioiden voidaan esittää sarakkeina. Dimension vähentämisen tavoitteena on vähentää näiden sarakkeiden määrää. Useasti on niin, että nämä mitat ovat korreloineet ja siksi datassa on joitain tarpeettomia tietoja. Tämä ylimääräinen tieto vaikuttaa heikentävästi koneoppimisen opettamiseen ja suorituskyykyyn, joten siksi dimensioiden vähentäminen on erittäin tärkeää. Menetelmässä voidaan toteuttaa ominaisuuksien valinta eli valitaan osajoukko alkuperäisen tietojoukon ominaisuuksista. Toinen tapa on ominaisuuksien erottaminen, jossa tietoa johdetaan alkuperäisjoukosta uuden ominaisuuden aliavaruuden rakentamiseksi. Tällöin tulosten ominaisuudet eivät ole enää samat kuin alkuperäiset. (Roman, 2019)

3.4 Vahvistusoppiminen

Vahvistusoppiminen on yksi koneoppimisen tekniikka, jonka avulla agentti oppii interaktiivisessa ympäristössä kokeilemalla ja erehtymällä, käyttäen palautetta omilta toimiltaan ja kokemuksiltaan. (Bhatt, 2018) Vahvistusoppimisessa oppiva kone tutkii ympäristöään ja jokaisesta tekemästä toimenpiteestä saadaan positiivista tai negatiivista palautetta. Tavoitteena on löytää ratkaisu, jolla saavutetaan eniten positiivista palautetta. Vahvistusoppimisessa ensimmäisen ratkaisun löytämisen jälkeen voidaan jatkaa vielä hakemalla uusia ja mahdollisesti parempia ratkaisuja. (Pietikäinen ja Silvén s. 69)

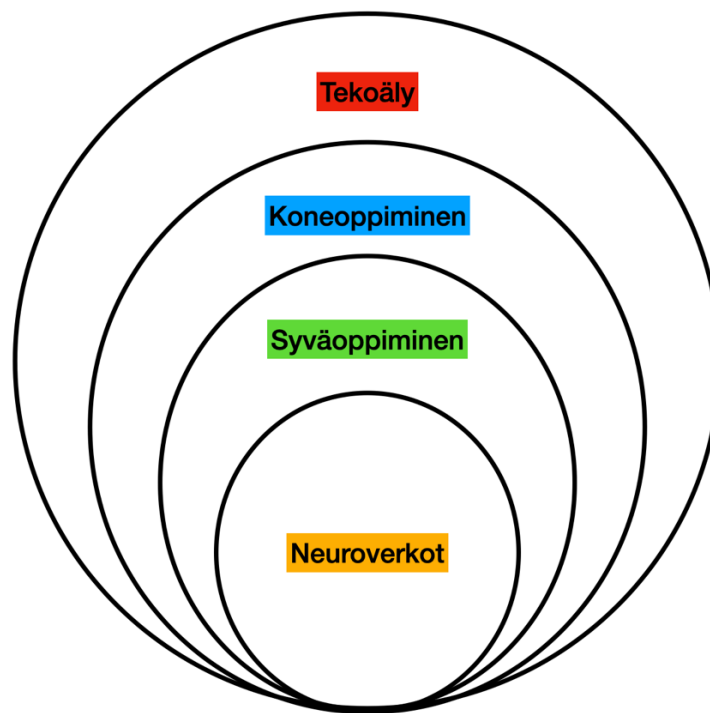
Vahvistusoppiminen on erilaista kuin ohjattu oppiminen, jossa opitaan ulkopuolisen valvontaohjelman tarjoamasta merkittyjen esimerkkien koulutusjoukosta. Vahvistusoppiminen eroaa myös ohjaamattomasta oppimisesta, jossa tyypillisesti on kyse

löytää rakenne piilotetusta datakokoelmista. Vahvistusoppimisen ongelmana on tutkimuksen ja hyödyntämisen välinen valinta ja suuri tarvittu palautteiden määrä sekä hidas oppiminen. Saavuttaakseen suuren määrän palkkiota, vahvistavan oppimisagentin on käytettävä toimia, joita se on kokeillut aiemmin ja joiden on todettu olevan tehokkaita palkkion tuottamisessa. (Sutton & Barto, 2017, s.3)

Eräs esimerkki vahvistusoppimisen sovelluksesta on mobiilirobotti, jonka tehtävänä on päättää, pitäisikö jatkaa seuraavaan huoneeseen etsimään lisää roskia kerättäväksi vai onko tarve etsiä reitti takaisin latausasemalle. Robotti tekee päätöksen akun nykyisen lataustason ja sen perusteella, kuinka nopeasti ja helposti se on aiemmin löytänyt laturin. Kuitenkaan toimintojen vaikutuksia ei voida täysin ennustaa, joten agentin on seurattava ympäristöään usein ja sen on reagoitava asianmukaisesti. (Sutton & Barto, 2017, s.4)

3.5 Neuroverkot ja syväoppiminen

Keinotekoiset hermoverkot (eng. Artificial Neural Network) eli neuroverkot muistuttavat ihmisen hermoverkkoja mutta näitä voidaan simuloida tietokoneen avulla. Ne koostuvat neuroneista eli hermosoluista, jotka lähettävät viestejä ja vastaanottavat niitä. Neuronit taas koostuvat solukeskuksesta ja johdoista, joiden avulla neuronit ovat kytkeytyneet toisiinsa. Syväoppiminen on koneoppimismenetelmä, jossa yksinkertaisista prosessointiyksiköistä koostuvia kerroksia yhdistetään verkostoksi siten, että järjestelmän prosessoima tieto kulkee niiden läpi vuoron perään. Neuroverkkojen erityispiirteenä voidaan pitää järjestelmän arkkitehtuuria, jossa jokainen neuroni voi prosessoida tietoa toisistaan riippumatta. Tällöin mahdollistetaan valtavien tietomäärien samanaikainen prosessointi. Perinteisesti tiedon tallennus tapahtuu tietokoneen muistissa ja CPU (eng. Central processing unit) käsittelee dataa. Neuroverkoissa neuronit sekä tallentavat että prosessoivat tietoa. (Reaktor & Helsingin yliopisto, 2018)

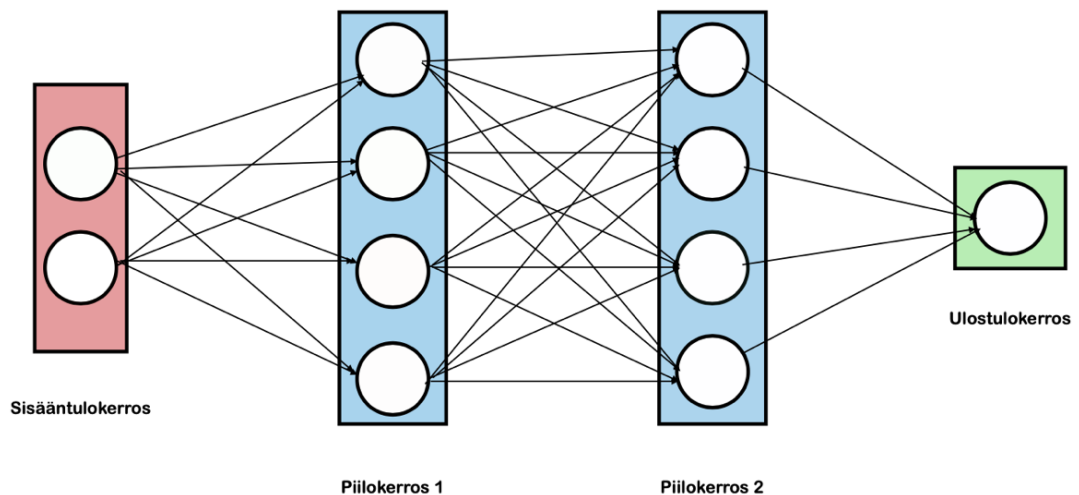


Kuva 2. Tekoälyn käsitteet (mukaillen Kavlakoglu, 2020).

Kuvassa 2 nähdään, mihin syväoppiminen ja neuroverkot sijoittuvat tekoälyssä kokonaisuudessaan. Koneoppiminen on tekoälyn osa-alue ja syväoppiminen on koneoppimisen osa-alue ja neuroverkot muodostavat syvällisten oppimisalgoritmien perustan. Syväoppimisesta tekee syvän neuroverkkorakenne eli miten monta kerrosta neuroverkossa on. Neuroverkko, joka koostuu enemmästä kuin kolmesta kerroksista, voidaan ajatella olevan syväoppimista. (Kavlakoglu, 2020)

Neuroverkot ovat adaptiivinen järjestelmä, joka oppii käyttämällä toisiinsa liittyviä solmuja tai neuroneja. Neuroverkko voi oppia datasta, joten sitä voidaan opettaa tunnistamaan kaavoja ja malleja, luokittelemaan tietoja ja ennustamaan tulevia tapahtumia. Ne koostuvat sisääntulokerroksesta, yhdestä tai useammasta piilokerroksesta ja ulostulokerroksesta kuvan 3 mukaisesti. Neuroverkot sopivat erityisen hyvin mallien tunnistamiseen esineiden tai signaalien tunnistamiseksi ja luokitteluksi puheesta, näöstä ja ohjausjärjestelmistä. Neuroverkkoja voidaan myös käyttää aikasarjojen ennustamisen ja mallintamisen esittämisessä. (Mathworks, 2020)

Kuvassa 3 yksinkertaistettu malli neuroverkosta, jossa piilokerroksia on kaksi kappaletta.



Kuva 3. Neuroverkko (mukaillen Dormehl, 2019).

Syväoppiminen vaatii suuren määrän merkittävää dataa. Esimerkiksi itseohjautuvan auton kehittäminen tarvitsee miljoonia kuvia ja tuhansia tunteja videoita toimiakseen. Syväoppiminen tarvitsee myös huomattavaa laskentatehoa. Tehokkailla grafiikkaprosessoreilla on rinnakkaisarkkitehtuuri, joka on tehokas syväoppimisessa. Tämä yhdistettynä klustereihin tai pilvipalveluihin, antaa kehitykselle mahdollisuuden lyhentää syvään oppivaan verkostoon liittyvää harjoittelu-aikaa viikoista jopa tunteihin. Teollisuudessa syväoppiminen auttaa esimerkiksi parantamaan työntekijöiden turvallisuutta raskaiden työstökoneiden lähellä havaitsemalla automaattisesti, kun ihmisiä tai esineitä on vaarallisella etäisyydellä koneista. Toinen esimerkki syväoppimisesta on automaattinen kuulon ja puheen kääntäminen. Esimerkiksi kotiavusteiset laitteet, jotka reagoivat puheeseen ja tietävät oikeat asetukset, toimivat syväoppimissovellusten avulla. (Mathworks, 2020)

3.6 Konenäkö

Konenäköjärjestelmät ovat joukko integroituja komponentteja, jotka on suunniteltu käyttämään digitaalisista kuvista otettua tietoa ohjaamaan automaattisesti valmistus- ja tuotantoprosesseja kuten testaus- ja laadunvalvontaprosesseja. (Edwards, 2020) Konenäön (eng. Machine vision) päätavoitteena on saada kone näkemään ja ymmärtämään mitä kamerasensorin kuvaama näkymä sisältää ja kuinka soveltaa tätä tietoa. (Ikeuchi K, 2014, Pietikäinen & Silvén mukaan, 2019, s. 127) Kuvan automaattinen tulkinta on hyvin vaativaa, sillä valaistuksen, katselusuunnan ja taustan

muutokset ja muut vaihtelut toimintaympäristössä aiheuttavat vaikeuksia kuvatus näkymän tulkinnalle. Yksinkertaisimmillaan kuvat ovat kaksiulotteisia, jolloin syvyystietoja ei tarvita. Kolmiulotteisen kuvan tulkinta on huomattavasti haastavampaa, sillä kohde näyttää erilaiselta riippuen tarkastelupisteestä. Konenäköjärjestelmä koostuu useista komponenteista, joita ovat valaisujärjestelmä, optinen järjestelmä tai linssi, anturi, näönkäsittelyjärjestelmä ja tiedonvälitysjärjestelmä. Se voi koostua erillisistä komponenteista tai ne voidaan integroida yhdeksi yksiköksi. Esimerkiksi älykamera, joka yhdistää yksittäisten elementtien toiminnot yhdeksi paketiksi.

Valaisujärjestelmä tulisi valita niin, että kontrasti maksimoidaan mitattavien ominaisuuksien kannalta, jotka halutaan havaita, ja puolestaan minimoidaan kaikkien muiden mitattavasta osasta. Käytettävän linssin tai optisen järjestelmän valinta riippuu konenäköjärjestelmän suorittamasta toiminnasta ja havainnoitavan ominaisuuden mitoista. Antureiden tehtävänä on auttaa kaappaamaan optisen järjestelmän valo ja muuttaa se digitaaliseksi kuvaksi. Suuremmalla resoluutiolla voidaan tuottaa kuvia, joissa on enemmän pikseleitä ja tämä parantaa kuvan laatua ja järjestelmän kykyä ratkaista yksityiskohtia. Anturin resoluutio riippuu havaittujen osien koosta, suoritettavien mittausten mitoista, mittausten toleransseista ja muista sovellusparametreista. (Edwards, 2020)

Esimerkkinä kaksiulotteisen, yksittäisen kuvan kuva-analyysiprosessi: Ensimmäinen vaihe on kuvan muodostus kuvalähteellä, joka voi olla digikamera. Tästä seuraa kuvan esikäsittely, jossa kuva muokataan analyysin kannalta sopivaan muotoon, esimerkiksi suodattamalla pois kohinaa tai muita sävyvaihteluita, jotka voivat haitata analyysiä. Tämän jälkeen kuva segmentoidaan, jossa tavoitteena on erottaa kohteet ja kohteiden osat toisistaan ja taustasta. Kriittisen segmentoinnin jälkeen lasketaan segmentoitujen alueiden ja reunojen ominaisuuksia kuvaavia piirteitä, esimerkiksi muotoa tai väriä. Analysoitavia kuvia verrataan systeemille etukäteen opetettujen prototyyppikohteiden malleihin ja pyritään tunnistamaan kuvassa esiintyviä kohteita tai löytämään poikkeamia. (Pietikäinen & Silvén, 2019, s. 130)

Konenäössä sovelletaan tietokonenäköä (eng. Computer vision) teolliseen tarkoitukseen esimerkiksi laadunvalvontaan tai robottinäköön. Konenäkö on siis tietokonenäön alaluokka. Erona näillä on se, että konenäköjärjestelmä vaatii tietokoneen ja tietyn ohjelmiston toimiakseen mutta tietokonenäköä ei tarvitse integroida koneeseen.

Tietokonenäkö voi esimerkiksi analysoida digitaalisia online kuvia tai videoita sekä kuvia liiketunnistimista, infrapunasensoreista tai muista lähteistä, ei siis ainoastaan normaaleista valokuvista tai videoista. (Marr, 2019)

4 TEKOÄLYN SOVELLUKSET KONETEKNIIKASSA

4.1 Betonin puristuslujuuden ennustaminen koneoppimisella

Sharma, Kumas & Chopra (Sharma, Kumar & Chopra, 2017) vertailevat artikkelissaan koneoppimisen tekniikoita betonin puristuslujuuden ennustamisessa. He käyttävät R-ohjelmistoympäristöä esitetyissä koneoppimistekniikoissa ja käytännön kokeet on luotu laboratorio-olosuhteissa. Tutkimuksessa betonin puristuslujuuden ennustamiseen on käytetty kolmea mallia: Päättöpuun malli (Decision tree model, DT), Satunnainen metsä-malli (Random forest model, RF) ja neuroverkkojen malli (Neural network, NN). Betoni on rakennusmateriaali, joka on sementin, veden, hienojen kiviainesten ja karkeiden kiviainesten seosta. Rakenteissa, joilta vaaditaan riittävä lujuus, voivat olla vaarallisia rakenteen vahvistussuunnittelussa esiintyvien virheiden vuoksi ja ympäristön vaikutus on myös suuri.

Päättöpuu-mallissa on taipumuksena mallintaa korkeamman asteen epälineaarisuutta ja suurempaa tutkittavuutta. DT- malli jakaa tiedot pienempiin alaryhmiin puurakenteen muodossa. Solmuja on kolmentyyppisiä, juurisolmu (eng. Root node), päätössolmu (eng. Decision node) ja lehtisolmu (eng. Leaf node). Ylimpänä on juurisolmu, päätössolmussa suoritetaan ehdollinen testi ja muut alipuut päätetään testin tuloksen perusteella. Lehtisolmu on puun kohdesolmu (ulostulo). DT-algoritmi laskee kahdentyyppiset entropiat: entropia, joka käyttää yhden määritteen taajuustaulukkoa ja entropia, joka käyttää kahden ominaisuuden taajuustaulukkoa. Seuraavaksi lasketaan entropian tai informaation vahvistuksen (eng. Gain) väheneminen. Lopuksi lasketaan tietojoukon kunkin haaran informaation vahvistus. Päätössolmuksi valitaan attribuutti, jolla on suurin informaation vahvistus ja haara, jolla vahvistus on 0, on lehtisolmu. Algoritmi toimii rekursiivisesti, itseään toistavasti, haaroissa, joissa ei ole lehtiä, kunnes kaikki tieto on luokiteltu.

Satunnaisen metsän malli on kokonaisluokittelu-menettelytapa. Ero DT:n ja RF:n välillä on se, että DT:ssä rakennetaan vain yksi puu, mutta RF-mallin tapauksessa useita. Kokonaisuusprosessin avulla päätellään lopullinen luokka. Metsäksi kutsutaan tätä joukkoa puita. Jokaiselle puulle on valittu satunnaisesti eri datakehys. Datakehys on

rivien ja sarakkeiden mitat ja jokaiselle puulle on valittu eri alaryhmä rivejä ja sarakkeita. Jokainen puu rakentuu seuraavilla vaiheilla: 1) Valitaan datakehys ja tämä datakehys on 66% puiden kokonaisharjoitusdatasta, jota kutsutaan pussiksi. Enne-muuttujat valitaan satunnaisesti ja näiden muuttujien parasta jakoa käytetään solmun jakamiseen.

2) Jokaiselle puulle lasketaan pakkausvirhe käyttämällä loput tiedosta. Sitten kaikkien puiden virheet kerätään kokonaispakkausvirhe-asteen löytämiseksi. 3) Jokainen puu antaa luokituksen ja RF-malli valitsee metsästä, jolla on suurin osa äänistä kaikista metsän puista. Äänet voivat olla 0 tai 1 ja saatujen 1 äänien prosenttiosuus määrittellään ennustustodennäköisyydeksi.

Neuroverkkomallin soveltamista on käytetty rakennussuunnittelun alalla jo yli kolmen vuosikymmenen ajan. NN koostuu seuraavista kolmesta pääelementistä: 1) Kerrosten rakenteesta ja niiden yhdistämisestä verkon muihin kerroksiin eli topologiasta. 2) Tekniikoista tietojen tallentamiseksi verkkoon eli oppimisesta ja 3) menetelmistä tietojen noutamiseksi verkosta eli muistista. Yleisesti NN-rakenne koostuu tulokerroksesta, yhdestä tai useammasta piilokerroksesta ja ulostulokerroksesta. Tässä tutkimuksessa otetut neuroverkot ovat myös yksi piilokerrosverkoista, joilla on yksi ulostulo. Tässä mainitussa työssä kehitetään kolmikerroksinen NN-malli, jonka avulla ennustetaan betonin puristuslujuus. 28 päivän puristuslujuuden ennustamiseen käytetään kolmea tuloa ja lähtö (output) on betonin 28 päivän puristuslujuus. Jos taas ennustetaan betonin 56 päivän puristuslujuutta, käytetään neljää tuloa missä yksi tulo on 28 päivän kovettuva puristuslujuus ja ulostulona on betonin 56 päivän puristuslujuus. 91 päivän puristuslujuuden ennustamiseen käytetään viittä tuloa, jossa yksi tuloista on 28 päivän kovettuva puristuslujuus ja toinen tuloista on 56 päivän kovettuva puristuslujuus ja ulostulona on betonin 91 päivän puristuslujuus.

Lopputulokseksi saadaan, että NN-malli ja RF-malli ovat ennustaneet hyvin lähelle betonin puristuslujuuden mitattuja arvoja. DT-malli ennusti parhaansa silloin kun lujuus on 91 päivän kovetusiässä. DT-mallin ongelmana on se, että se ei pysty ennustamaan harjoittelutietojoukossa annetun alueen ulkopuolelle. Tässä tutkimuksessa parhaan ennusteen teki neuroverkko eli NN- malli, joka ennusti suurella tarkkuudella kaikissa tapauksissa, 28, 56 ja 91 kovettumispäiville. Yksi neuroverkkojen tärkeimmistä eduista verrattuna RF ja DT-malleihin on, että NN pystyy arvioimaan erittäin epälineaariset toiminnot tulo-lähtöominaisuuksien välillä eikä vaadi ennakkotietoa kyseisen suhteen

luonteesta. RF-malli tulee heti NN-mallin jälkeen betonin 56 ja 91 päivän puristuslujuudelle mutta se vie enemmän aikaa kuin NN. DT-malli ei ole pystynyt ennustamaan betonin puristuslujuutta nykyiselle aineistolle. Myös R ohjelmistoympäristö todettiin hyödylliseksi ohjelmoijille ja muille tutkijoille tietojen tilastolliseen analysointiin. (Sharma, Kumar & Chopra, 2017)

4.2 Konenäkö laaduntarkastuksessa ja kunnossapidossa - yleistä

Konenäön avulla voidaan tehostaa tuotannon valmistusprosesseja auttamalla valmistajia ennakoimaan tuotantolinjojen tuotantovirheitä, parantamaan laatua, vähentämään tarpeettomia kustannuksia ja saavuttamaan korkea tuottavuus automaation avulla. Tuotannon suuria koneita tai raskaita laitteita on seurattava säännöllisesti ja niille on suoritettava kunnossapitoa, jotta vältetään turhat seisakit. Manuaalinen tarkastus tuotantolaitoksessa on aikaa vievää ja kustannuksiltaan korkeaa ja virhealtista. Tämän takia pyritään ennakoimaan koneiden toimintatilaa ja ryhdytään jo ennakoivasti korjaaviin toimiin, jotta vältetään todennäköisesti tapahtuvilta vioilta, kuten laitteen kulumisilta ja likaantumisilta. Robottien avulla voidaan ottaa kuvia tarkasteltavista laitteista, jonka jälkeen kuvat ja laitteen lisätiedot syötetään tietokoneelle, jossa analyysi ja käsittely tapahtuu. Näin valmistaja saa tiedon siitä, onko kyseinen laite vikaantumassa vai ei.

Konenäköjärjestelmillä teolliset valmistajat voivat havaita fyysisen tuotteen puutteita, halkeamia tai vikoja. Lisäksi järjestelmillä voidaan tarkistaa komponenttien tai osien tarkat mitat jo kokoonpanon aikana. Konenäköjärjestelmä ottaa kuvan tuotteesta ja kuvat lähetetään tietokoneelle käsittelyä varten. Näin voidaan toteuttaa automaattista tuotetarkastusta ja tarkkaa laadunvalvontaa. (Joshi, 2019)

4.3 Konenäkö tarkastusta varten - tapaustutkimus

Miles ja Surgenor (Miles & Surgenor, 2011) esittelevät tapaustutkimuksessaan autoteollisuuden tapausta, jossa automatisoitua tarkastusjärjestelmää käytetään havaitsemaan kiinnittimen asettamisen oikeellisuus. Samalla verrataan kuuden eri luokittelijan suorituskkyä puuttuvien kiinnittimien havaitsemiseen. Tässä otetaan huomioon vain konenäköjärjestelmän asettelu, ohjelmisto ja opetettava datajoukko. Tutkittavat luokittelijatyypit olivat hermoverkkopohjainen prosessori

pääkomponenttianalyysi tulo-ominaisuusjoukon luokittelemiseksi uudelleen ja suora Eigenimage-lähestymistapa välttämään tarvetta poimia ominaisuuksia jokaisesta kuvasta.

Tutkimuksessa oleva osa on auton poikittaispalkki, joka on metallinen tuki ja sijaitsee auton kojelaudan takana. Palkki on meistetty metalliosa ja meistetty metallinen radion kannatin hitsataan kiinni tähän palkkiin. Kokoonpanoprosessissa kiinnitetään neljä pientä suorakulmaista pidikettä radiokannattimeen. Nämä neljä pidikettä kiinnittävät radioyksikön poikittaispalkkiin. Tämä on turvallisuuden kannalta kriittinen osa ja on välttämätöntä varmistaa, että pidikkeet on asennettu oikein, sillä törmäyksessä radionyksikön on oltava tukevasti kiinni. Pidikkeiden läsnäolon varmistamiseksi kehitettiin konenäköjärjestelmä, joka tarkastaa jokaisen osan automaattisesti ennen kuin se poistuu kiinnittimille tarkoitetusta valmistussolusta. PLC, joka ohjaa valmistussolua, on yhteydessä erillisen tietokoneen kanssa, joka ohjaa näköjärjestelmää.

Tuotantolinjalle asennettiin kaksi kamerajärjestelmää, joilla tehdään osien tarkastus. Valaistuksena on diffuusorilla varustettu LED-rengasvalo. Tietojen luokittelussa käytettiin MATLAB -pohjaista QVision ohjelmistojärjestelmää. Ohjelmistolla pystytään lataamaan kirjastokuva, valitsemaan ja poimimaan Eigenimagesina kiinnostavia ominaisuuksia tai alueita ja kouluttamaan luokittimia uusien kuvien lopulliseen tarkastukseen asti. Ohjelmisto integroitiin PLC:hen, joka ylläpitää valmistussolua osien online-tarkastusta varten.

Luokitustekniikoiden vertailussa käytettiin kahta erilaista aineistoa. Ne sisältävät läpimenneitä ja hylättyjä kuvia osista kokoonpanolinjalla. Hyväksytty kuva koostuu olemassa olevista kiinnittimistä ja hylätyissä kuvissa puuttuvista kiinnikkeistä. Ylä- ja alakiinnittimiä kohti on asetettu kaksi kameraa, joten kuviakin muodostuu kaksi. Ensimmäinen datajoukko, jota kutsutaan alkuperäiseksi joukoksi, koostui 150 hyväksytystä kuvasta ja 114 hylätystä kuvasta ylemmässä haarukassa ja 147 hyväksytystä ja 125 epäonnistuneesta kuvasta alemmassa. Toinen datajoukko, jota kutsutaan suuntajoukoksi, on tutkinut järjestelmän suorituskyvyn määrittämistä kahdessa olosuhteessa. Valaistusta on muutettu ja lisäksi on otettu kolme uutta vikatilaa käyttöön. Vikaantumistapoja ovat puuttuva pidike, taaksepäin ja ylösalaisin oleva pidike, taaksepäin oleva pidike sekä vielä ylösalaisin oleva pidike. (Miles & Surgenor, 2011)

4.4 Materiaalien löytö ja suunnittelu koneoppimisella

Liu ym. (2017) tutkivat artikkelissaan menetelmiä, joilla uusien materiaalien löytö- ja suunnitteluprosessi nopeutuisi. Perinteiset kokeet ja laskennallinen mallinnus kuluttavat usein valtavan määrän aikaa ja resursseja ja niiden kokeelliset olosuhteet ja teoreettiset perusteet rajoittavat usein niitä. Materiaalien löytämiseen ja suunnitteluun koneoppimisen avulla on kiinnitetty yhä enemmän huomiota ja ne ovat saavuttaneet suuria parannuksia sekä aikatehokkuudessa että ennustustarkkuudessa.

Katsauksessaan Liu ym. (2017) esittävät koneoppimisen perusmenetelmät materiaalitieteessä ja tutkivat tämänhetkistä tutkimustilannetta koneoppimisen sovellusten suhteen materiaalien ominaisuuksien ennustamisessa ja uusien materiaalien löytämisessä. Tarkoituksena on löytää parametreit, jotka vaikuttavat materiaalien ominaisuuksiin ja mahdollistavat tehokkaamman tutkimuksen materiaalien löytämisessä ja suunnittelussa.

Materiaalitekniikassa yleisesti käytetyt koneoppimisalgoritmit voidaan jakaa neljään luokkaan, jotka ovat todennäköisyysarvio, regressio, klusterointi ja luokittelu. Erityisesti todennäköisyysarviota käytetään pääasiassa uusien materiaalien löytämiseen seulomalla erilaisia rakenteiden ja komponenttien yhdistelmiä. Kun taas regressiota, klusterointia ja luokittelua käytetään aineellisten ominaisuuksien ennustamiseen makro- ja mikrotasolla. Koneoppimisen kolme perusvaihetta materiaalitieteissä ovat näytteen rakenne, mallin rakentaminen ja lopuksi mallin arvioiminen. Ensimmäisessä vaiheessa kerätään alkuperäiset tiedot laskennallisista simulaatioista ja kokeellisista mittauksista rajataan vain ehdolliset tekijät sekä päätösominaisuuksien kannalta olevat merkityksettömät tekijät jätetään pois. Tämän jälkeen luodaan ehdollisten tekijöiden ja päätösominaisuuksien välisten suhteiden malli annetun otoksen perusteella. Tietopohjaisen mallin tulisi saavuttaa hyvä suorituskky olemassa olevalla ja myös näkymättömällä datalla. Lopuksi mallien yleistysvirheet arvioidaan laskentaperusteisten testien avulla ja valitaan tulosten perusteella paras malli. Tutkimus esittää kaksi tapaa aineellisten ominaisuuksien ennustamiseen: makroskooppisen suorituskyyvyn ennusteen ja mikroskooppisen ominaisuuksien ennusteen.

Materiaalien makroskooppisen suorituskyyvyn tutkimus keskittyy lähinnä materiaalin makroskooppisten esimerkiksi mekaanisten ja fysikaalisten ominaisuuksien ja mikrorakenteen väliseen rakenneaktiivisuus-suhteeseen. Regressioon ja luokitteluun liittyvät ongelmat ratkaistaan hyvin, joten koneoppimismenetelmiä, joissa käytetään

ANN (eng. Artificial neural network) -ja SVM (eng. Support vector machine) algoritmeja yhdessä optimointialgoritmien kanssa, on käytetty laajasti makroskooppisen suorituskyvyn ennustamiseen tutkimuksessa. Erästä ANN:in sovellusta on käytetty ennustamaan materiaalikäyttäytymistä kuten lämpötilan muutos ja vetolujuus, venymä, hukka, korroosio ja puristusominaisuudet. Nämä ominaisuudet ovat hyvin tärkeässä osassa konetekniikassa. SVM-tekniikka on SLT (eng. Statistical learning theory) -pohjainen ohjattu oppimismenetelmä tietojen analysoimiseksi ja luokittelussa ja regressiossa käytettävien mallien tunnistamiseksi. ANN- malleihin verrattuna SVM-mallit puolestaan soveltuvat paremmin pieniin otoksiin ja ne voivat ratkaista ylikuormituksen ja ulottuvuuden ongelmat.

Materiaalin makroskooppinen suorituskky määräytyy sen mikroskooppisten ominaisuuksien, mukaan lukien atomien ja rakenteellisten ominaisuuksien, kuten hilavakion, mukaan. Koneoppimisen sovellukset mikroskooppisessa ominaisuusennusteessa keskittyvät yleensä useisiin näkökohtiin kuten hilavakio, kaistaenergia, elektroni-affiniteetti ja molekyyli-atomisaationenergia. Nopeasta ja luotettavasta ratkaisusta suuren määrän tuntemattomien yhdisteiden hilavakioiden ennustamiseksi on tulossa keskeinen vaatimus korkean teknologian kehittämisessä. Koneoppimista voidaan soveltaa tämän ennustusongelman ratkaisemiseen sen hyvän käytännöllisyyden vuoksi regressioanalyysissä.

Koneoppimisen yleinen prosessi uusien materiaalien löytämisessä koostuu oppimisjärjestelmästä ja ennustusjärjestelmästä. Oppimisjärjestelmä suorittaa tietojen puhdistuksen, ominaisuuksien valinnan ja mallikoulutuksen sekä testauksen. Ennustusjärjestelmä käyttää mallia, joka saadaan oppimisjärjestelmästä komponenttien ja rakenteiden ennustamiseen. Tällä hetkellä uusien materiaalien löytämiseen liittyvät kokeelliset ja laskennalliset seulonnot sisältävät elementtien korvaamisen ja rakenteen muunnoksen. Uusien materiaalien löytäminen voidaan jakaa kahteen menettelytapaan, jotka keskittyvät kiteiden rakenteen ennustamiseen ja koostumuksen ennustamiseen keskittyviin menetelmiin. Koneoppimista käytetään laajemmin komponenttiennusteissa kuin kiteiden ennustamisessa. Kiteiden ennustaminen voidaan suorittaa regressioanalyysin avulla ilman ennakkotietoa, mutta komponenttiennuste etenee ratkaisemalla jälkeen päin tulevat todennäköisyydet Bayesin tilastomallin avulla. Bayesin tilastomalli kiinnittää huomiota paitsi otoksen kokonaistietojen ja yksittäisten otosten tietojen käyttöön myös ennakkotietojen keräämiseen, louhintaan ja käsittelyyn. (Liu ym., 2017)

4.5 Pyörivien koneiden vianmääritys tekoälyn avulla

Pyörivät koneet ovat tärkeimpiä laitteita nykyaikaisissa teollisissa sovelluksissa ja niiden vianmäärityksestä tulee kriittisin osa järjestelmän suunnittelussa ja huollossa. Vianmäärityksellä on kolme perustehtävää: määritetään, onko laite normaali vai ei, alkavan murtumisen ja sen syyn löytäminen sekä vikakehityksen kehityssuunnan ennustaminen. Vastesignaalien vaihtelevuuden ja runsauden vuoksi vikakuvioita on mahdotonta tunnistaa suoraan. Siksi vianmääritysjärjestelmä koostuu usein kahdesta keskeisestä vaiheesta: tietojenkäsittely, jossa poimitaan ominaisuudet ja vian tunnistaminen. Älykkäät vianmääritysjärjestelmät on rakennettu muuntamaan tulomallit siten, että ne voidaan esittää matalaulotteisilla ominaisuusvektoreilla helpomman vastaavuuden ja vertailun varmistamiseksi. (LeCun yms. 1998, Liu yms. mukaan, 2018).

Ominaisuusvektoreita käytetään AI-tekniikoiden tulona vian tunnistamiseen. Vian tunnistuksen vaihe tarkoittaa ominaisuusalueella saatujen tietojen kartoittamista koneen vikoihin vikatilassa. Erityisesti luokittelijoita ja tilastollisia oppimismenetelmiä on käytetty laajalti pyörivän koneen vianmäärityksessä, joka sisältää k-lähimmän naapurin algoritmit, Bayesin luokittelijan, tukivektorikoneen ja keinotekoisen neuroverkon. Myös syväoppimisen lähestymistapoja on alettu soveltaa vianmäärityksen alalla. Nämä tekoälyn algoritmit pyörivien koneiden vianmääritykseen ovat tulleet suosituiksi niiden kestävyys ja sopeutumiskyvyn ansiosta. Datan fuusiointia sekä piirteiden ulottuvuuden vähentämistekniikoita on käytetty laajasti teollisuusjärjestelmissä. Esimerkiksi J. K. Sinha yms. (2013) ovat ehdottaneet tapaa vähentää ääriäänäntureiden määrää edistyneillä signaalinkäsittely- ja fuusiointitekniikoilla. Tämä puolestaan voi vähentää riippuvuutta käytännön sovellusten kokemuksista ja subjektiivisista arvioista. (Liu yms., 2018)

k-NN on osoitettu toimivan hyvin ja sitä on sovellettu laajalti vianmäärityksessä. Ilmentymäpohjaisena algoritmina k-NN:ää voidaan käyttää sekä luokitteluun että regressioon. k-NN:n avainkysymys on k:n arvon valinta, mikä voi vaikuttaa merkittävästi algoritmin suorituskäytävyyteen ja joka tulisi valita huolellisesti todellisissa sovelluksissa.

Naiivi Bayes on generatiivinen malli, jolla on korkea oppimisen ja ennustamisen tehokkuus ja joka on helppo toteuttaa. (S.B. Kotsiantis yms. 2007, Ruonan Liu yms. mukaan, 2018) Ehdollisen riippumattomuuden vahvan oletuksen ja ennakkotiedon tarpeen takia naiivi Bayes-menetelmä soveltuu vain itsenäiselle piirteisvektorille.

Teolliset signaalitoiminnot ovat usein riippuvaisia toisistaan. Siksi naiivisten Bayes-sovellusten toteutus tapahtuu usein jonkin dimensioiden vähentämistoimenpiteen tai valkaisevan esikäsittelyn jälkeen.

SVM:ssä eli tukivektorikone-luokittelumallissa ytimen valinta on tärkeä ja vaikuttaa suoraan näytteiden erottavuuteen ja luokittelutehoon. SVM:ää käytetään yleensä yhdessä signaalinkäsittelytekniikoiden kanssa ominaisuuksien poimimiseksi.

ANN:illa eli neuroverkolla on voimakas malliluokittelu ja kuvioiden tunnistuskyky. Tämän takia ANN on yksi luokittelijoista, joita käytetään yleisimmin älykkäissä vikadiagnoosissa. Kuten SVM, myös ANN:n sovellus on yhdessä asianmukaisten ominaisuuksien poimijoiden kanssa. Käytännössä ANN-malliin lisätään usein laillistustermi tai ennakkotieto, jotta vältetään liiallinen sovitus ja saavutetaan parempi tarkkuus.

Syväoppimismenetelmät perustuvat laskennallisten elementtien syvään arkkitehtuuriin. Klassinen ja yleinen esimerkki tällaisesta elementistä on ANN, jota voidaan käyttää rakentamaan syvä neuroverkko syvällä arkkitehtuurilla. (Y. Bengio 2009, Liu yms mukaan 2018) Käytännössä syvän arkkitehtuurin takia parametrien määrä kasvaa, mikä johtaa ylisovittamisen riskiin, jota yritetään välttää esimerkiksi laillistamisen avulla.

4.6 Mitä uutta tekoäly tuo konetekniikkaan?

Tekoäly mahdollistaa sen, että tulevaisuudessa ihmiset voivat työskennellä entistä tiiviimmin koneiden kanssa. Koneoppimisen avulla varmistetaan, että koneet pystyvät yhä enemmän tunnistamaan ihmisten tai prosessin tarpeet ja pystyvät vastaamaan niihin. Tulevaisuudessa voidaan monet toistuvat tehtävät, kuten laaduntarkastukset, toteuttaa tai tukea tekoälypohjaisilla ratkaisuilla. (Reimann, 2020) Tekoäly auttaa konetekniikan prosessin aikana erilaisilla tavoilla. Esimerkiksi 3D-rakenteiden jännityksen arviointi arvioi jännityksen 3D-rakenteiden suunnittelussa ja valmistuksessa. Toinen esimerkki on materiaalien arviointi eli materiaalin lujuuden, kestävyyyden ja laadun arviointi. Rakenteen luomisessa tekoäly voi auttaa algoritmiensa ja tietojen tallennuksensa kautta. Se tekee prosessista tehokkaan ja läpinäkyvän.

Konetekniikassa käytetään CAD-järjestelmää mekaniikkasuunnittelussa. Tekoäly tuo muutoksia suoraan siihen, mitä CAD voi tehdä. Ohjelmiston muodossa oleva älykkyys muuntaa CAD-järjestelmän iteratiivisesta työkalusta generatiiviseksi avustajaksi. Generatiivisella suunnittelulla tarkoitetaan sitä, että laskentatehoa hyödynnettäisiin ottamalla paras olemassa oleva ratkaisu ongelmaan ja toistamalla optimoidaan suorituskyky tietyssä ympäristössä. Tällä hetkellä huomattava osa ajasta kuluu geometrian kokoamiseen, mutta tekoäly voi ottaa haltuunsa geometrian ja tämä mahdollistaa suunnittelijoille aikaa muuhun työhön. Kun tekoäly ja generaattori huolehtivat perusasioista, voidaan käyttää enemmän aikaa luovuuden käyttämiseen. (English, 2018)

Tekoäly vaikuttaa tuotesuunnitteluun ja suunnittelujärjestelmien luomiseen kolmessa erillisessä vaiheessa tulevan vuosikymmenen aikana. Ensinnäkin keinotekoisesti älykkäät järjestelmät helpottavat suunnittelijoiden kohtaamia työläitä tehtäviä kuten etsimällä jatkuvasti sopivaa sisältöä, korjaamalla virheitä, määrittämällä optimaaliset ratkaisut, ilmoittamalla muutoksista ja seuraamaan suunnittelun epäonnistumisia. Koneoppiminen pystyy ottamaan vastaan nämä tehtävät ja tekemään ne paljon nopeammin. Seuraavaksi tekoäly voi auttaa kehittämään monimutkaisia malleja. Älykkäät järjestelmät toimivat suunnittelijan apuna ja ehdottavat erilaisia vaihtoehtoja, sisällyttämällä anturipohjaisia tietoja, luomalla suunnittelun edeltäjiä ja optimoimalla toimitusketjuprosesseja ja toimittamalla mallit älykkäisiin tuotantolaitoksiin. Viimeisenä vaiheena olisivat tekniset järjestelmät, jotka sisältävät vahvemman tekoälyn ja voivat toimia enemmän kuin ihmisen avustajina suunnittelun ja luomisen aikana. Älykkäät järjestelmät ja koneet toimivat tulosten suuntausten mukaisesti ja luovat uudet suunnittelun iteraatiot tarkistettavaksi. (Noor, 2017)

5 YHTEENVETO

Tavoitteena oli tutustua tekoälyyn ja sen sovelluksiin konetekniikassa. Työssä tekoälyä ja sen osa-alueita esiteltiin käsitteitä ja niiden merkitystä läpikäymällä ja esittämällä esimerkkejä. Tekoälyn luokittelua erilaisiin ryhmiin ja alaryhmiin tehtiin työssä, jotta kokonaiskuva tekoälystä voitaisiin muodostaa. Suurimpana osana tekoälyä keskityttiin koneoppimiseen, jota käytetään myös esitetyissä sovelluksissa. Koneoppimiseen liittyy erilaisia algoritmeja, joita esitellään työssä ja sovelluksissa.

Esitellyissä esimerkeissä koneoppimista sovellettiin betonin puristuslujuuden määrittämiseen ja vertailtiin kolmea eri luokittelumallia. Näistä NN- malli eli hermovekko malli osoittautui parhaaksi sen kykynsä takia arvioida epälineaariset toiminnot tulo-lähtöominaisuuksien välillä. Toinen sovellus oli tapaustutkimus autoteollisuudesta, jossa konenäöllä luotiin automatisoitu tarkastusjärjestelmä. Tapaustutkimuksessa tarkastusjärjestelmää käytettiin havaitsemaan kiinnittimen oikea asennus. Konenäön avulla tarkistettiin, oliko tarkasteltu osa oikeassa, halutussa asennossa. Samassa tapaustutkimuksessa esiteltiin käytännössä, mistä osista konenäköjärjestelmä koostui ja kuinka dataa kerättiin. Sovelluskohteena oli myös materiaalien ominaisuuksien ennustaminen ja uusien materiaalien löytäminen koneoppimisen avulla. Artikkelissa etsittiin sopivat parametrit, jotka vaikuttavat materiaalien ominaisuuksiin ja näin voitiin tehostaa materiaalien tutkimusta jatkossakin. Yksi sovelluksista oli myös vianmääritys. Pyörivien koneiden vianmäärityksessä käytettiin luokittelijoita ja tilastollisia oppimismenetelmiä, jotka ovat k-lähimmän naapurin algoritmit, Bayesin luokittelija, tukivektorikone ja keinotekoinen neuroverkko.

Tekoäly ja koneoppiminen konetekniikan alalla mahdollistaa tulevaisuudessa entistä tiiviimmän ihmisen ja koneen välisen yhteistyön. Tekoälypohjaisten ratkaisujen käyttö ei korvaa ihmisiä konetekniikassa, mutta edellyttää heitä kouluttajina tai esimiehinä. Tekoälyllä tulee olemaan yhä suurempi merkitys konetekniikan alalla kohdistuen eri vaiheisiin, suunnittelusta kunnossapitoon, koko tuotteen elinkaaren ajan.

LÄHDELUETTELO

Bhatt, S. 2018. 5 Things you need to know about reinforcement learning [verkkodokumentti]. Saatavissa: <https://www.kdnuggets.com/2018/03/5-things-reinforcement-learning.html> [viitattu 5.10.2020]

Brownlee, J. 2016. Supervised and Unsupervised Machine Learning Algorithms [verkkodokumentti]. Australia: Machine Learning Mastery. Saatavissa: <https://machinelearningmastery.com/supervised-and-unsupervised-machine-learning-algorithms/> [viitattu 1.10.2020]

Chopra, T., Kumar, M., Kumar Sharma, R., 2018. "Comparison of Machine Learning Techniques for the Prediction of Compressive Strength of Concrete", Advances in Civil Engineering, vol 2018, ID 5481705, 9 s. Saatavissa: <https://www.hindawi.com/journals/ace/2018/5481705/> [viitattu 6.10.2020]

CGI, 2020. Mitä on tekoäly? – Hyödynnä tekoälyä liiketoiminnassa [verkkodokumentti]. Helsinki: CGI Suomi Oy. Saatavissa: <https://www.cgi.fi/fi/mita-on-tekoaly> [viitattu 21.9.2020].

Dickson, B. 2019. What is symbolic artificial intelligence? [verkkodokumentti]. Saatavissa: <https://bdtechtalks.com/2019/11/18/what-is-symbolic-artificial-intelligence/> [viitattu 28.9.2020]

Dormehl, L. 2019. What is an artificial neural network? Here's everything you need to know [verkkodokumentti]. UK: Digital Trends. Saatavissa: <https://www.digitaltrends.com/cool-tech/what-is-an-artificial-neural-network/> [viitattu 28.10.2020]

Edward, E. 2020. An introduction to machine vision systems [verkkodokumentti]. New York: Thomas. Saatavissa: <https://www.thomasnet.com/articles/automation-electronics/machine-vision-systems/> [viitattu 6.10.2020]

Elements of AI, 2020. Elements of AI, Koneoppimisen lajit [verkkodokumentti]. Helsinki: Reaktor & Helsingin yliopisto. Saatavissa: <https://course.elementsofai.com/fi/4/1> [viitattu 28.9.2020]

Elements of AI, 2020. Elements of AI, Neuroverkkojen periaatteet [verkkodokumentti]. Helsinki: Reaktor & Helsingin yliopisto. Saatavissa: <https://course.elementsofai.com/fi/4/1> [viitattu 28.10.2020]

English, T. 2018. What AI and automation means for the future of mechanical engineering? [verkkodokumentti] Autodesk: Inventor blog. Saatavissa: <https://blogs.autodesk.com/inventor/2018/02/12/ai-automation-means-future-mechanical-engineering/> [viitattu 31.10.2020]

FabricAI, 2020. Mitä on tekoäly? [verkkodokumentti]. Tampere: FabricAI Oy. Saatavissa: <https://fabricai.fi/mita-on-tekoaly/> [viitattu 26.9.2020]

Gandhi, R. 2018. Support vector machine – Introduction to machine learning algorithms [verkkodokumentti]. Saatavissa: <https://towardsdatascience.com/support-vector-machine-introduction-to-machine-learning-algorithms-934a444fca47> [viitattu 28.10.2020]

Gupta, P. 2017. Naive Bayes in Machine Learning [verkkodokumentti]. Saatavissa: <https://towardsdatascience.com/naive-bayes-in-machine-learning-f49cc8f831b4> [viitattu 28.10.2020]

Harrison, O. 2018 Machine Learning basics with the k-nearest neighbors algorithm [verkkodokumentti]. Saatavissa: <https://towardsdatascience.com/machine-learning-basics-with-the-k-nearest-neighbors-algorithm-6a6e71d01761> [viitattu 28.10.2020]

Joshi, N., 2019. 4 Machine vision use cases for the manufacturing industry [verkkodokumentti]. Navi Mumbai: Allerin. Saatavissa: <https://www.allerin.com/blog/4-machine-vision-use-cases-for-the-manufacturing-industry> [viitattu 8.10.2020]

Kavlakoglu, E., 2020. AI vs. Machine learning vs. deep learning vs. neural networks: What's the difference [verkkodokumentti]. New York: IBM. Saatavissa: <https://www.ibm.com/cloud/blog/ai-vs-machine-learning-vs-deep-learning-vs-neural-networks> [viitattu 5.10.2020]

Liu, R., Yang, B., Zio, E. & Chen, X. 2018, “Artificial intelligence for fault diagnosis of rotating machinery: A review”, *Mechanical Systems and Signal Processing*, vol. 108, ss. 33-47. Saatavissa: <https://www.sciencedirect.com/science/article/abs/pii/S0888327018300748> [viitattu 28.10.2020]

Liu, Y., Zhao, T., Ju, W. & Shi, S. 2017, “Materials discovery and design using machine learning”, *Journal of Materiomics*, vol 3, no. 3, ss. 159-177. Saatavissa: <https://www.sciencedirect.com/science/article/pii/S2352847817300515?via%3Dihub> [viitattu 12.10.2020]

Marr, B., 2019. What is machine vision and how is it used in business today? [verkkodokumentti]. New Jersey: Forbes. Saatavissa: <https://www.forbes.com/sites/bernardmarr/2019/10/11/what-is-machine-vision-and-how-is-it-used-in-business-today/?sh=305f2d606939#1dc965566939> [viitattu 6.10.2020]

Mathworks, 2020. What is deep learning? 3 things you need to know [verkkodokumentti]. Saatavissa: <https://www.mathworks.com/discovery/deep-learning.html> [viitattu 5.10.2020]

Mathworks, 2020. What is a neural network? 3 things you need to know [verkkodokumentti]. Saatavissa: https://www.mathworks.com/discovery/neural-network.html?s_tid=srchtitle [viitattu 5.10.2020]

Merilehto, A., 2018. Tekoäly - Matkaopas johtajalle. Helsinki: Alma Talent, 206 s. ISBN 9789521433344.

Miles, B. & Surgenor, B., 2011. Machine vision for inspection: A case study, Assembly line – theory and practice, Waldemar Grzechca, IntechOpen. Saatavissa: <https://www.intechopen.com/books/assembly-line-theory-and-practice/machine-vision-for-inspection-a-case-study> [viitattu 10.10.2020]

Noor, A.K. 2017. "AI and the future of the machine design." ASME. Mechanical engineering. vol. 130 (10) ss. 38-43. Saatavissa: <https://asmedigitalcollection.asme.org/memagazineselect/article/139/10/38/380304/AI-and-the-Future-of-the-Machine> [viitattu 31.10.2020]

Pietikäinen, M., Silvén, O. 2019. Tekoälyn haasteet – Koneoppimisesta ja konenäöstä tunnetekoälyyn [verkkodokumentti]. Oulun yliopisto: Koneäön ja signaalianalyysin keskus, 253 s. ISBN 978-952-62-2482-4 (elektroninen julkaisu) [Viitattu 21.9.]. Saatavilla: <http://jultika.oulu.fi/Record/isbn978-952-62-2482-4>

Reimann, G. 2020. Artificial intelligence – the future of mechanical engineering [verkkodokumentti]. Saksa: VDMA. Saatavissa: <https://www.vdma.org/en/v2viewer/-/v2article/render/36900765> [viitattu 31.10.2020]

Roman, V. 2019. Unsupervised learning: dimensionality reduction [verkkodokumentti]. Saatavissa: <https://towardsdatascience.com/unsupervised-learning-dimensionality-reduction-ddb4d55e0757> [viitattu 3.10.2020]

Sutton, R., Barto, A. 2017. Reinforcement Learning: An Introduction. [verkkodokumentti] Saatavissa: <http://incompleteideas.net/book/bookdraft2017nov5.pdf> 2 painos. [viitattu 5.10.2020]. 427 s.

Tuominen, H., Neittaanmäki, P., Lehto, M., Niinimäki, E., Nyrhinen, R., Ojalainen, A., Pölönen, I., Rautiainen, I., Ruohonen, T., Vähäkainu, P., Äyrämö, S., Äyrämö, S.M. 2019. Tekoälyn perusteita ja sovelluksia [verkkodokumentti]. Jyväskylä: Jyväskylän yliopisto [viitattu 28.9.2020] Saatavilla: <https://tim.jyu.fi/view/kurssit/tie/tiep1000/tekoalyn-sovellukset/kirja#DKUvbnUuGytQ>